Preparing the SDK-51 to Emulate the 8052

Preparing the SDK-51 to emulate the 8052's 16-bit timer and additional RAM is a fairly straight forward task. Once you follow the steps below, you can use the SDK-51 to fully emulate all the features of the 8052 except the additional 8K of internal ROM.

Step 1

Replace the 8051 chip in the SDK-51 with an 8052 chip.

Step 2

Use a PROM programmer to copy the monitor firmware into a buffer or a disk file. The monitor PROM to be changed is the left-hand one of the two-PROM set.

Step 3

The SDK-51's standard monitor firmware doesn't allow you to access the 8052's additional RAM using the DBYTE command. If you try to do so, you get the message:

ERR 12 ADDR OUT OF RANGE

To change the allowed range and the method of addressing the extra data RAM, you need to change the monitor routine (I) FETCH/(I)STORE. To do this, change the hexadecimal values of the monitor code you copied in Step 2 from addresses E6AC to E6C9 to:

B4,01,08, 85,45,82, 20,D5,E6, C1,97,B4, 02,15,E5, 45,30,E7, F0,F8,20, D5,03,E6, C1,9D,EA, F6,C1,9D

(The spaces between the groups of hexadecimal numbers are there only to improve readability.)

Figure 2 shows the instructions for the original code at this location and the replacement code.

Step 4

The original monitor software also does a checksum on the two monitor proms. To make the stored checksum value compatible with the new code, change the value at address E049H to 8DH.

Step 5

Burn the modified code into a new PROM. Replace the original monitor PROM with the new PROM.

This article is one of several that will appear in the next update of the ICE-51 Technical Report.

Highlighting iRMX-86™ System Booting

Software Support Services has answers for iRMX-86 release 5 users that face complications when attempting to boot their system. This information and other aspects of iRMX-86 is covered in the iRMX-86 Technical Report to be published in Q4/83.

We have explained what is happening as you are booting your system in the chart below. The symptom is described with the associated cause and solution detailed below it. Check each cause carefully to pin-point your actual situation.

Symptom: When attempting to boot the system there is disk access, but no console output and no response to keyboard input.

CAUSE	SOLUTION
1. Early versions of LIB-86 could produce invalid load files.	1. Use only LIB-86 version 2.1 or later.
2. The 8086 processor is waiting for a response from the 8087 Numeric Data Processor.	2. If the 8087 is not present in the system, be sure the Test Pin (number 23) of the 8086 is tied to ground.
3. Automatic bootstrap device recognition does not function when booting from a release 4 FORMATTED disk. The second stage of the bootstrap loader was changed in release 5 to allow for Auto Boot support.	3. Reformat using the release 5 FORMAT command or FILES Utility if necessary.
4. The second stage of the bootstrap loader is being overwritten when loading the operating system.	4. Do not locate the operating system or user written applications in the same physical location as the second stage of the bootstrap loader. Refer to the "Guide to Installing the iRMX-86 release 5 Operating System on System 86/300 Series Microcomputer Systems".
5. The device name in the bootstrap loader does not have a corresponding device name in a DUIB.	5. Using the ICU, make sure that the device name of the booting device exists in a DUIB. ("WO" for the Winchester disk)
6. A user job(s) was configured into the system, but the jobs object module was not added into the bootable library. This may commonly occur when using any of the Intel provided definition files for the System 86/3xx series products. The SDB is configured as a user job, but has not been added to the bootable library.	6. Add the user jobs (the SDB) object modules to the bootable library using the LIB-86 command.
7. There is a conflict over an interrupt, (two interrupts are assigned to the same level).	7. Check both the board jumpers and the ICU configuration for two interrupts assigned to the same level.
8. The bootstrap loader does not use interrupts. The iRMX-86 Operating System does use interrupts.	8. Check that the interrupt is actually getting to the 8259A.
9. In a system configured with the iSBC-544, the dual port RAM on the iSBC-544 board is configured as system memory.	9. During configuration (using the ICU), exclude the ISBC-544 memory from the memory made available to iRMX-86 Operating System. Also, if using an 86/330 or 86/380 system, the physical memory must end 16K bytes before the beginning of the iSBC-544 RAM.
10. The system console is configured for automatic baud rate detection.	10. Press SHIFT U to determine the baud rate.
11. In a system configured with a user job or a user I/O job, an incorrect start address has been specified.	11. Check the Load and Start Addresses. They are not necessarily the same, especially when using the LARGE model. During ICU configuration, ensure that the start address is specified.
12. A first level user job has a missing or unexecuted ENDSINITSTASK system call.	12. Check that all first level user jobs invoke an ENDSINITSTASK system call.
13. A user I/O job has an ENDSINITSTASK system call.	13. User I/O jobs cannot invoke ENDSINITSTASK system calls.
14. In a system configured with the iSBC-208 board, the I/O base address of the iSBC-208 was changed for release 5.	14. Reconfigure the bootstrap loader EPROM to reflect the release 5 I/O Base Address of 180H.
15. In a system configured with the iSBC-254S Bubble Memory Board, a patch is missing.	15. Apply patch number 95. Note that patch number 93 is incorrect and should not be used.

New Revision of User's Guide

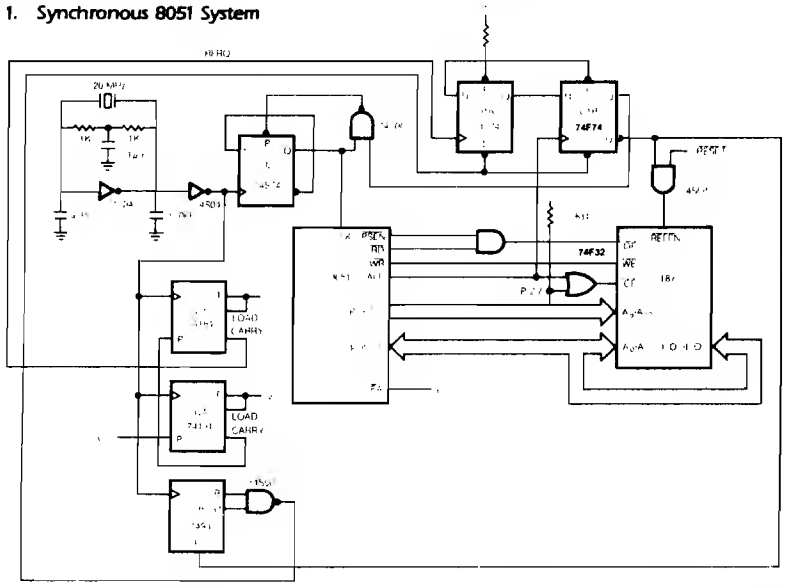
There are improvements in the next revision of the NDS-II ISIS-II (N) User's Guide Supplement that we would like you to know about. One improvement regards a clarification of volume and directory names used in Figure 2-2 (in the present manual) and their corresponding text references. These can be found on pages 2-2 and 2-3. Also, the REMOVE command will be added to the list on page 4-1 of the ISIS-II (N) commands associated with NDS-II. The next version will be revision 004 and is currently being produced.

ICE™-51 and the 2187A

There has been some concern that the ICE-51 does not work properly with the Intel 2187A byte-wide RAM. The problem is not with the ICE-51 or the 2187A; it is a problem with the component family chosen for several 2187 circuits described in the Intel literature.

The circuit described in the application note "Designing Memory Systems With the 8K x 8 iRAM" (AP-132) exceeds the drive specification at the ALE line. In figure 26 of this application note, ALE is tied to two Schottky gates (74S32 and 74S74) requiring an IOL of 4 mA (2 gates x 2 mA). ALE is only able to drive an IOL of 3.2mA. The use of these parts may cause memory failure with the ICE-51. The 74S32 and 74S74 parts should be replaced with **74F32** and **74F74** parts. The F family parts provide the advantages of high speed and lower drive requirements. With the use of the F family parts, ICE-51 should work without failure. The corrected circuit diagram is shown in Figure 1 below.

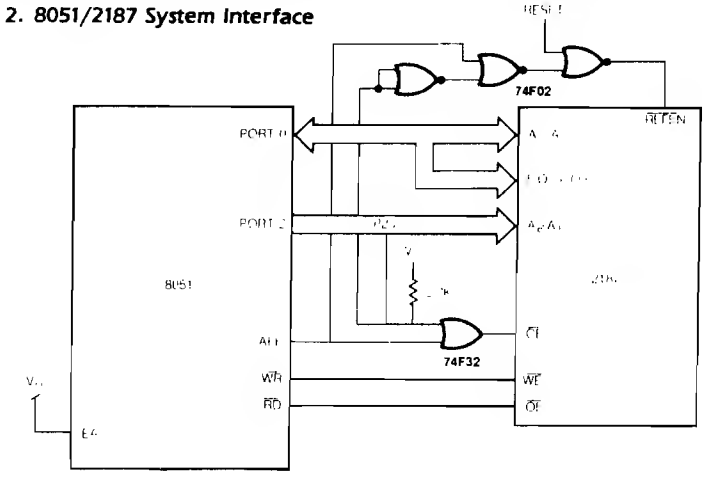
Figure 1. Synchronous 8051 System



In the application brief "The Designers Guide to iRAMs", the microcontroller example for the 2187 shows 74S02 and 74S32 parts. As stated above, the use of these parts exceeds the drive capability of ALE. The F family parts, **74F02** and **74F32**, should be used instead.

The 2187A data sheet has an error in one of the circuit diagrams. Figure 1 of the data sheet, showing the 8051 and 2187 system interface, indicates a NOR gate is to be used, but the gate is labeled as a 74F32, which is an OR gate. The part should be labeled as **74F02.** The corrected diagram, as it should appear in the data sheet and application brief, is shown in Figure 2.

Figure 2. 8051/2187 System Interface



For Your Information

The information detailed below is a result of some of the Software Performance Reports (SPRs) submitted to Software Support Services, as well as other information we thought would be useful to you. Several situations are outlined below to share the engineering remedies with you.

In this article, several products are reviewed, including FORTRAN-86, FORTRAN-80, LOC-86, and ASM-86.

Fortran-86 Version 2.1

Here is a situation you may not know about. Incorrect data is sometimes passed to a function when its actual argument is a multiple-segment or dummy-array reference. The problem exists when the multiple-segment or dummy-array reference appears more than once in the expression. For example:

```
A = FUNC1(X(I)) + FUNC2(X(I))
```

where FUNC1 and FUNC2 are functions.

To avoid this problem, use a temporary variable, as follows:

```
TEMP = X(I)
100 A = FUNC1(TEMP) + FUNC2(TEMP)
```

or

```
TEMP = FUNC1(X(I))
100 A = TEMP + FUNC2(X(I))
```

Another situation to be aware of is that you can run into incorrect code when a 16-Bit Multiply is immediately followed by a 16-Bit Divide in the same expression. Detour this by assigning the result of the multiplication to a 16-Bit intermediate variable, which in turn is used in subsequent labeled statements.

Have you encountered incorrect results? They can be generated when a 16-Bit MOD function reference is followed by a division in the same expression. This happens when the 16-bit MOD function needs to be coerced to a 32-bit value. You will most likely see this occur for an expression of the form **MOD (I)/J**. A workaround would be to use a 16-Bit intermediate variable as follows:

```
ITEMP = MOD(I)
100 N = ITEMP/J
```

If you use CONTROL Z to stop reading from the console, you may get an 'EOF' error message. To avoid this, you need to rewind (REWIND(5)). The error message is issued because a flag is set to show an end-of-file when a CONTROL Z is entered. REWIND resets the flag and allows you to continue reading.

The following error message has been omitted from the FORTRAN-86 User's Guide:

1300H INVALID LINK SEQUENCE

You'll be pleased to note that the next version of FORTRAN-86 is planned for release in Q4 of 1983.